



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/800,122	03/12/2004	Nobuo Togahara	4142.70026	9096
24978	7590	06/11/2007	EXAMINER	
GREER, BURNS & CRAIN			LU, KUEN S	
300 S WACKER DR				
25TH FLOOR			ART UNIT	PAPER NUMBER
CHICAGO, IL 60606			2167	
			MAIL DATE	DELIVERY MODE
			06/11/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	10/800,122	TOGAHARA, NOBUO	
	Examiner	Art Unit	
	Kuen S. Lu	2167	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) Responsive to communication(s) filed on 29 March 2007.
- 2a) This action is **FINAL**. 2b) This action is non-final.
- 3) Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) Claim(s) 1-21 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) Claim(s) _____ is/are allowed.
- 6) Claim(s) 1-21 is/are rejected.
- 7) Claim(s) _____ is/are objected to.
- 8) Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) The specification is objected to by the Examiner.
- 10) The drawing(s) filed on _____ is/are: a) accepted or b) objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
 - a) All b) Some * c) None of:
 1. Certified copies of the priority documents have been received.
 2. Certified copies of the priority documents have been received in Application No. _____.
 3. Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|--|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413) |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | Paper No(s)/Mail Date. _____ |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO/SB/08) | 5) <input type="checkbox"/> Notice of Informal Patent Application |
| Paper No(s)/Mail Date .. | 6) <input type="checkbox"/> Other: _____ |

DETAILED ACTION

1. The Action is responsive to Applicant's Amendment filed on March 29, 2007.

Applicant's Amendment amending claims 1-2, 6-8, 10, 13 and 15-21 is acknowledged. Also acknowledged is amendment made to abstract.

Examiner's objections to specification and abstract, and rejections to claims 1-2, 4, 6-7 and 14, 16-17 and 19-20 under 35 U.S.C. § 101 is hereby withdrawn. Please note claims 8-9, 15, 18 and 21 remain rejected under 35 U.S.C. § 101.

2. Please note claims 1 and 2-21 are pending.

3. As to Applicant's Arguments/Remarks filed March 29, 2007, please see Examiner's response in "Response to Arguments**", following this Office Action for Final Rejection (hereafter "the Action"), shown next.**

Claim Rejections - 35 USC § 101

4. 35 U.S.C. 101 reads as follows:

Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

4.1. As set forth in MPEP 2106 (II) (A):

The claimed invention as a whole must accomplish a practical application. That is, it must produce a "useful, concrete and tangible result." State Street, 149 F.3d at 1373, 47 USPQ2d at 1601-02. The purpose of this requirement is to limit patent protection to inventions that possess a certain level of "real world" value, as opposed to subject matter that represents nothing more than an idea or concept, or is simply a starting point for future investigation or research (Brenner v. Manson, 383 U.S. 519, 528-36, 148 USPQ 689, 693-96); In re Ziegler, 992, F.2d 1197, 1200-03, 26 USPQ2d 1600, 1603-06 (Fed. Cir. 1993)). Accordingly, a complete disclosure should contain some indication of the practical application for the claimed invention, i.e., why the applicant believes the claimed invention is useful.

Apart from the utility requirement of 35 U.S.C. 101, usefulness under the patent eligibility standard requires significant functionality to be present to satisfy the useful result aspect of the practical application

Art Unit: 2167

requirement. See *Arrhythmia*, 958 F.2d at 1057, 22 USPQ2d at 1036. Merely claiming nonfunctional descriptive material stored in a computer-readable medium does not make the invention eligible for patenting. For example, a claim directed to a word processing file stored on a disk may satisfy the utility requirement of 35 U.S.C. 101 since the information stored may have some "real world" value. However, the mere fact that the claim may satisfy the utility requirement of 35 U.S.C. 101 does not mean that a useful result is achieved under the practical application requirement. The claimed invention as a whole must produce a "useful, concrete and tangible" result to have a practical application.

4.2. As set forth in MPEP 2106 (IV) (B) (1):

Claims to computer-related inventions that are clearly nonstatutory fall into the same general categories as nonstatutory claims in other arts, namely natural phenomena such as magnetism, and abstract ideas or laws of nature which constitute "descriptive material." Abstract ideas, *Warmerdam*, 33 F.3d at 1360, 31 USPQ2d at 1759, or the mere manipulation of abstract ideas, *Schrader*, 22 F.3d at 292-93, 30 USPQ2d at 1457-58, are not patentable. Descriptive material can be characterized as either "functional descriptive material" or "nonfunctional descriptive material." In this context, "functional descriptive material" consists of data structures and computer programs which impart functionality when employed as a computer component. (The definition of "data structure" is "a physical or logical relationship among data elements, designed to support specific data manipulation functions." The New IEEE Standard Dictionary of Electrical and Electronics Terms 308 (5th ed. 1993).) "Nonfunctional descriptive material" includes but is not limited to music, literary works and a compilation or mere arrangement of data. Both types of "descriptive material" are nonstatutory when claimed as descriptive material *per se*. *Warmerdam*, 33 F.3d at 1360, 31 USPQ2d at 1759. When functional descriptive material is recorded on some computer-readable medium it becomes structurally and functionally interrelated to the medium and will be statutory in most cases since use of technology permits the function of the descriptive material to be realized. Compare *In re Lowry*, 32 F.3d 1579, 1583-84, 32 USPQ2d 1031, 1035 (Fed. Cir. 1994) (claim to data structure stored on a computer readable medium that increases computer efficiency held statutory) and *Warmerdam*, 33 F.3d at 1360-61, 31 USPQ2d at 1759 (claim to computer having a specific data structure stored in memory held statutory product-by-process claim) with *Warmerdam*, 33 F.3d at 1361, 31 USPQ2d at 1760 (claim to a data structure *per se* held nonstatutory).

4.3. As set forth in MPEP 2106 (IV)(B)(1)(a):

Similarly, computer programs claimed as computer listings *per se*, i.e., the descriptions or expressions of the programs, are not physical "things." They are neither computer components nor statutory processes, as they are not "acts" being performed. Such claimed computer programs do not define any structural and functional interrelationships between the computer program and other claimed elements of a computer which permit the computer program's functionality to be realized. In contrast, a claimed computer-readable medium encoded with a computer program is a computer element which defines structural and functional interrelationships between the computer program and the rest of the computer which permit the computer program's functionality to be realized, and is thus statutory. Accordingly, it is important to distinguish claims that define descriptive material *per se* from claims that define statutory inventions.

Products may be either machines, manufactures, or compositions of matter.

A machine is "a concrete thing, consisting of parts or of certain devices and combinations of devices. *Burr v. Duryee*, 68 U.S. (1 Wall.) 531, 570 (1863).

If a claim defines a useful machine or manufacture by identifying the physical structure of the machine or manufacture in terms of its hardware or hardware and software combination, it defines a statutory product. See, e.g., *Lowry*, 32 F.3d at 1583, 32 USPQ2d at 1034-35; *Warmerdarn*, 33 F.3d at 1361-62, 31 USPQ2d at 1760.

Office personnel must treat each claim as a whole. The mere fact that a hardware element is recited in a claim does not necessarily limit the claim to a specific machine or manufacture. Cf. *In re Iwahashi*, 888 F.2d 1370, 1374-75, 12 USPQ2d 1908, 191 1-12 (Fed. Cir. 1989), cited with approval in *Alappat*,

Art Unit: 2167

33 F.3d at 1544 n.24, 31 USPQ2d at 1558 n_24.

4.5. Claims 8-9, 15, 18 and 21 are rejected under 35 U.S.C. § 101 because the claimed invention is directed to non-statutory subject matter.

As per claims 8, 18 and 21, the claims are method comprising, program embedded on medium to cause computer to perform and apparatus comprising components to perform, respectively, steps of obtaining, specifying and executing. It is noted the executing step is an intermediate step for producing some result to be utilized. However, there is no utility specified after the executing step. Therefore, performing the steps does not produce any useful result. However, a tangible, concrete and useful result is required in a practical application test. The consequence is non-statutory.

As per claims 9 and 15, the claims inherit the deficiency of being non-statutory directly or indirectly from claim 8, and do not remedy the deficiency individually or by inheritance. The consequence is non-statutory.

Claim Rejections - 35 USC § 103

5. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

This application currently names joint inventors. In considering patentability of the claims under 35 U.S.C. 103(a), the examiner presumes that the subject matter of the various claims was commonly owned at the time any inventions covered therein were made absent any evidence to the contrary. Applicant is

advised of the obligation under 37 CFR 1.56 to point out the inventor and invention dates of each claim that was not commonly owned at the time a later invention was made in order for the examiner to consider the applicability of 35 U.S.C. 103(c) and potential 35 U.S.C. 102(e), (f) or (g) prior art under 35 U.S.C. 103(a).

7.1. Claims 1-21 are rejected under 35 U.S.C. 103(a) as being unpatentable over Worden (U.S. Patent Application 2003/0149934) in view of Chandra et al. (U.S. Patent Application 2002/0138582, hereafter “Chandra”).

As per claims 1, 16 and 19, Worden teaches “An information processing method” (See Abstract where a program uses a set of mappings between XML and business information logical structures), “A program embodied on a medium, for causing a computer to execute an information processing, said program” (See [0054] where program is used to specify instructions to a computer) and “An information processing apparatus” (See [0002] where computer systems conducts e-business transactions), comprising:

“(an analyzer for) analyzing XML data corresponding to a form screen, and specifying a business class” … “corresponding to a tag included in said XML data” (See [0592] where a business model entity is represented by an XML element, an entity structure is represented by structures inside of the element and all entities of a given class are represented by elements of a given tag name).

Although Worden teaches object oriented programming at [0054], Worden does not distinct programming languages for application embodiment at [0323] and does not explicitly teach that the business class “encoded in an object oriented programming language”.

However, Chandra teaches the business class "encoded in an object oriented programming language" (See [0291] where specific appearance, contents and functions of application are defined by building blocks which are implemented in the forms of classes, methods and interfaces in object oriented programming).

It would have been obvious to one having ordinary skill in the art at the time of the applicant's invention was made to combine the teaching of Chandra with Worden reference by encoding business classes in object oriented language because Worden implements XML-based language requiring manipulating message content in terms of classes, attributes and relations of a business model while Chandra implements application in the form of programming object in an object oriented programming environment, the combined teaching would have enabled Worden's application comprising business classes in the form of object oriented containers such that Worden's business application would have been created and stored as transportable applications.

The combined teaching of the Worden and Chandra references further teaches the following:

"(a checker for) referring to a business class manager in which items of said business classes to be created are registered, and judging whether an item of the specified business class has not been registered in said business class manager" (See Chandra: [0259], [0291] and [0402] where specific appearance, contents and functions of business application classes are defined by building blocks which are created in the

forms of classes, methods and interfaces in object oriented programming, and registered in the system); and

“upon detecting that said item of the specified business class has not been registered in said business class manager, registering said item of the specified business class in said business class manager” and “a register for registering said item of the specified business class in said business class manager, if said item of the specified business class has not been registered in said business class manager” (See Chandra: [0259], [0291] and [0402] where specific appearance, contents and functions of business application classes are defined by building blocks which are created in the forms of classes, methods and interfaces in object oriented programming, and registered in the system, and Worden: [0197] where source code is generated by substituting class names in a standard template).

As per claim 2, the combined teaching of the Chandra and Worden references further teaches “The information processing method as set forth in claim 1, further comprising:
upon detecting that said item of the specified business class has not been registered in said business class manager, generating template source program data for the specified business class” (See Worden: [0229] and : [0197] where Java objects are created and source code is generated by substituting class names in a standard template, and Chandra: [0259], [0291] and [0402] where specific appearance, contents and functions of business application classes are defined by building blocks which are created in the forms of classes, methods and interfaces in object oriented programming,

and registered in the system).

As per claim 3, Worden teaches "The information processing method as set forth in claim 1, further comprising: reading out an HTML file for said form screen, and generating XML data corresponding to said form screen according to a predetermined rule" (See Worden: [0003] where XML is generated, viewed *and transforming into HTML file*; and further at [0050] where XML is derived from SGML in which HTML is an application).

As per claim 4, the combined teaching of the Chandra and Worden references further teaches "The information processing method as set forth in claim 1, wherein said analyzing and specifying comprises specifying a pre-processing class, a post-processing class and a form processing class, so as to correspond to a start tag or an end tag of said XML data corresponding to said form screen" (See Worden: [0229] where nodes and paths are defined by specifying one node and its type from root node, specifying two node types and a path to represent properties of an object of processing class and specifying three node types and paths to represent association between object classes).

As per claim 5, the combined teaching of the Chandra and Worden references further teaches "The information processing method as set forth in claim 3, further comprising: generating said HTML file for said form screen in response to an instruction of a user"

(See Worden: [0003] where XML is generated, viewed and transforming into HTML file).

As per claim 6, Worden teaches the information processing method as set forth in claim 1, further comprising:

“specifying a form item storing object by a tag included in said XML data corresponding to said form screen” (See Worden: [0592] where a business model entity is represented by an XML element, an entity structure is represented by structures inside of the element and all entities of a given class are represented by elements of a given tag name);

“referring to a form item storing object manager in which items of form item storing objects to be created are registered, and judging whether an item of the specified form item storing object has not been registered in said form item storing object manager”

(See Chandra: [0259], [0291] and [0402] where specific appearance, contents and functions of business application classes are defined by building blocks which are created in the forms of classes, methods and interfaces in object oriented programming, and registered in the system); and

“upon detecting that said item of the specified form item storing object has not been registered in said form item storing object manager, registering said item of the specified form item storing object into said form item storing object manager” (See Chandra: [0259], [0291] and [0402] where specific appearance, contents and functions of business application classes are defined by building blocks which are created in the forms of classes, methods and interfaces in object oriented programming, and

registered in the system).

As per claims 7, 17 and 20, Worden teaches “An information processing method” (See Abstract where a program uses a set of mappings between XML and business information logical structures), “A program embodied on a medium, for causing a computer to execute an information processing, said program” (See [0054] where program is used to specify instructions to a computer) and “An information processing apparatus” (See [0002] where computer systems conducts e-business transactions), comprising:

“(an analyzer for) analyzing XML data corresponding to a form screen, and specifying a business class corresponding to a tag included in said XML data” (See [0592] where a business model entity is represented by an XML element, an entity structure is represented by structures inside of the element and all entities of a given class are represented by elements of a given tag name).

“(an analyzer for) analyzing XML data corresponding to a form screen, and specifying a business class” ... “corresponding to a tag included in said XML data” (See [0592] where a business model entity is represented by an XML element, an entity structure is represented by structures inside of the element and all entities of a given class are represented by elements of a given tag name).

Although Worden teaches object oriented programming at [0054], Worden does not distinct programming languages for application embodiment at [0323] and does not

explicitly teach that the business class "encoded in an object oriented programming language".

However, Chandra teaches the business class "encoded in an object oriented programming language" (See [0291] where specific appearance, contents and functions of application are defined by building blocks which are implemented in the forms of classes, methods and interfaces in object oriented programming).

It would have been obvious to one having ordinary skill in the art at the time of the applicant's invention was made to combine the teaching of Chandra with Worden reference by encoding business classes in object oriented language because Worden implements XML-based language requiring manipulating message content in terms of classes, attributes and relations of a business model while Chandra implements application in the form of programming object in an object oriented programming environment, the combined teaching would have enabled Worden's application comprising business classes in the form of object oriented containers such that Worden's business application would have been created and stored as transportable applications.

The combined teaching of the Worden and Chandra references further teaches the following:

"(a checker for) referring to a business class manager in which items of said business classes to be created are registered, and judging whether an item of the specified business class has not been registered in said business class manager" (See Chandra: [0259], [0291] and [0402] where specific appearance, contents and functions of

business application classes are defined by building blocks which are created in the forms of classes, methods and interfaces in object oriented programming, and registered in the system); and

“upon detecting that said item of the specified business class has not been registered in said business class manager, generating template source program data for the specified business class” and “a generator for generating template source program data for the specified business class, if said item of the specified business class has not been registered in said business class manager” (See Chandra: [0259], [0291] and [0402] where specific appearance, contents and functions of business application classes are defined by building blocks which are created in the forms of classes, methods and interfaces in object oriented programming, and registered in the system, and Worden: [0197] where source code is generated by substituting class names in a standard template).

As per claims 8, 18 and 21, “An information processing method” (See Abstract where a program uses a set of mappings between XML and business information logical structures), “A program embodied on a medium, for causing a computer to execute an information processing, said program” (See [0054] where program is used to specify instructions to a computer) and “An information processing apparatus” (See [0002] where computer systems conducts e-business transactions), comprising: “obtaining XML data including a tag corresponding to data inputted or selected for a form screen, and specifying a business class that corresponds to said tag included in

said XML data and is a program for carrying out a processing relating to said tag" and "a first processor, obtaining XML data including a tag corresponding to data inputted or selected for a form screen, and specifying a business class that corresponds to said tag included in said XML data and is a program for carrying out a processing relating to said tag" (See [0592] and [0427] where a business model entity is represented by an XML element, an entity structure is represented by structures inside of the element and all entities of a given class are represented by elements of a given tag name and object of entity, attribute or relation is selected from a screen).

Although Worden teaches loading tree of business model onto screen at [0439] which involves CPU processing, but it does not seem specifying explicitly enough on teaching "executing the specified business class", based on Applicant's response filed on March 29, 2007.

However, Chandra teaches "executing the specified business class" by encoding and executing business class in an object oriented programming language (See [0291] and [0202] where specific appearance, contents and functions of application are defined by building blocks which are implemented in the forms of classes, methods and interfaces in object oriented programming and executing the building blocks).

It would have been obvious to one having ordinary skill in the art at the time of the applicant's invention was made to combine the teaching of Chandra with Worden reference by encoding business classes in object oriented language because Worden implements XML-based language requiring manipulating message content in terms of classes, attributes and relations of a business model while Chandra implements

Art Unit: 2167

application in the form of programming object in an object oriented programming environment, the combined teaching would have enabled Worden's application comprising business classes in the form of object oriented containers such that Worden's business application would have been created and stored as transportable applications.

As per claim 9, Worden teaches "The information processing method as set forth in claim 8, wherein said obtaining and specifying comprises specifying a pre-processing class, a post-processing class, and a form processing class that correspond to a form of said form screen and a tag included in said XML data" (See Worden: [0229] where nodes and paths are defined by specifying one node and its type from root node, specifying two node types and a path to represent properties of an object of processing class and specifying three node types and paths to represent association between object classes).

As per claim 10, Worden teaches the information processing method as set forth in claim 8, further comprising:
"outputting said data inputted or selected for said form screen to a form item storing object that is defined in advance and is loaded into a memory" (See Fig. 9-11 and [0239] where MDL extracts each component of meaning from input document and packages into output document);
"storing said data inputted or selected for said form screen into said memory by said

form item storing object" (See Fig. 9-11 and [0239] where MDL extracts each component of meaning from input document and packages into output document); and "performing a processing by exchanging data between the called business class and said form item storing object" (See Fig. 9-11 and [0239] where MDL extracts each component of meaning from input document and packages into output document and [0342] where messages are exchanged between applications and businesses).

As per claim 11, Worden teaches the information processing method as set forth in claim 10, wherein said performing comprises:

"if data is transferred from a first business class to a second business class, outputting said data to an interclass interface object that is defined in advance and is loaded into said memory, by said first business class" (See [0342] where messages are exchanged between applications and businesses); and
"referring to said interclass interface object and receiving said data from said interclass interface object by said second business class" (See [0342] and [0385] where messages are exchanged between applications and businesses, and relation is established between classes).

As per claim 12, Worden teaches the information processing method as set forth in claim 8, further comprising:

"receiving data inputted or selected for said form screen from an apparatus that displayed said form screen, and generating said XML data including said data inputted

or selected for said form screen and corresponding tags" (See [0592] where a business model entity is represented by an XML element, an entity structure is represented by structures inside of the element and all entities of a given class are represented by elements of a given tag name).

As per claim 13, the combined teaching of Worden and Chandra references further teaches "the information processing method as set forth in claim 10, further comprising: "if an output request is received from the executed business class, generating output XML data by using data held in said form item storing object" (See Worden: [0197] where source code is generated by substituting class names in a standard template, and Chandra: [0291] and [0202] where specific appearance, contents and functions of application are defined by building blocks which are implemented in the forms of classes, methods and interfaces in object oriented programming and executing the building blocks); and "outputting said output XML data to said apparatus that displayed said form screen" (See Worden: Fig. 9-11 and [0239] where MDL extracts each component of meaning from input document and packages into output document).

As per claim 14, Worden teaches "The information processing method as set forth in claim 8, further comprising: if the specified business class does not exist, generating and outputting error information" (See Fig. 9-11 and [0239], [0859] and [0938] where MDL extracts each component of meaning from input document and packages into

output document and error messages are created).

As per claim 15, the combined teaching of Worden and Chandra references further teaches "The information processing method as set forth in claim 8, wherein each of the called business classes is configured so as to complete a processing for the entire form relating to said form screen by a processing executed by the called business classes without a program defining a processing sequence" (See [0042] where mapping is established between an XML language onto a business model to read in data from XML and convert into an internal form reflecting business model structures for allowing data input and output between XML languages, and Chandra: [0291] and [0202] where specific appearance, contents and functions of application are defined by building blocks which are implemented in the forms of classes, methods and interfaces in object oriented programming and executing the building blocks).

Prior Art

6. The prior art made of record

- A. U.S. Patent Application 2003/0149934
- F. U.S. Patent Application 2002/0138582

The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

- B. U.S. Patent Application 2004/0006527
- C. U.S. Patent Application 2003/0233249
- D. U.S. Patent Application 2003/0014441

E. U.S. Patent 6,938,079

Response to Arguments

7. Applicant's arguments, filed on March 29, 2007, have been fully considered and please see discussion below.

At Page 12, Applicant traverses Examiner's rejections to claims 1-2, 4, 6-9 and 14-21 under 35 U.S.C. § 101. Based on amended claims, Examiner respectfully agrees to withdraw the rejections to all claims, excepting 8-9, 15, 18 and 21. For the rejections to claims 8-9, 15, 18 and 21, please see corresponding section in the Action.

At Page 13, concerning 35 U.S.C. § 102 and 103 rejections to claims 1-21 and Applicant's amendment made to claims 1-2, 6-8, 10, 13 and 15-21 which introduces business class encoded in an object oriented programming language and executing business class, Examiner respectfully introduced a new Chandra reference for providing teaching to the newly introduced subject matter in the amendment. Please see appropriate sections in the Action.

Conclusion

8. Applicant's amendment necessitated the new grounds of rejection presented in this Office Action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within

TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1 .136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Contact Information

9. Any inquiry concerning this communication or earlier communications from the Examiner should be directed to Kuen S. Lu whose telephone number is (571) 272-4114. The examiner can normally be reached on Monday-Friday (8:00 am-5:00 pm). If attempts to reach the examiner by telephone pre unsuccessful, the examiner's Supervisor, John Cottingham can be reached on (571) 272-7079. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for Page 13 published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll free). If you would like assistance from a

USPTO Customer Service Representative or access to the automated information system, please call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Kuen S. Lu, *WSL*

Patent Examiner, Art Unit 2167

June 6, 2007